

情報数学

第11回 表示の意味

萩野 達也

hagino@sfc.keio.ac.jp

Slides URL

<https://vu5.sfc.keio.ac.jp/slide/>

再帰関数の不動点意味論(例)

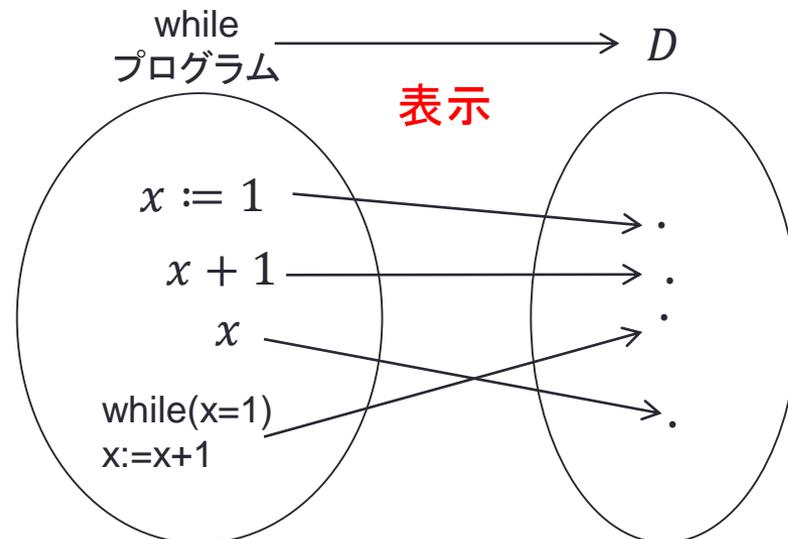
- 次の再帰的関数が何を計算するかを求めなさい.
 - $f(x) \equiv \text{cond}(x > 100, x - 10, f(f(x + 11)))$

プログラミング言語の意味

- プログラミング言語の構文
 - BNF(あるいは文脈自由文法)を使って形式的に定義されることが多い
- プログラミング言語の意味
 - 自然言語で説明されることが多く, あいまいである
- 形式的意味
 - 公理的意味
 - プログラムを論理式の中に埋め込む
 - 操作的意味
 - プログラムを別の良くわかっているシステムで模倣する
 - 表示的意味
 - プログラムを数学的対象として埋め込む

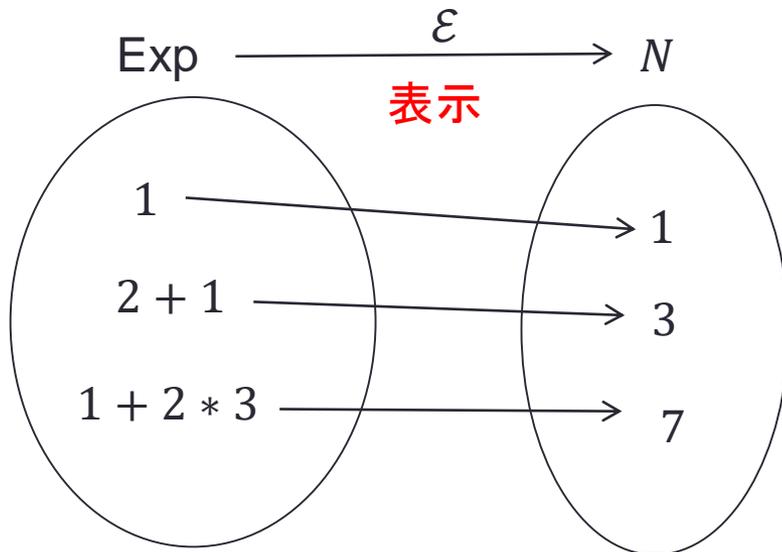
Whileプログラム

- Whileプログラム
 - $\text{input}(x_1, x_2, \dots, x_n)$
 - $\text{output}(y)$
 - $x := e$
 - $\{P_1; P_2; \dots; P_n\}$
 - $\text{if}(e_1 = e_2) \text{ then } P \text{ else } Q$
 - $\text{while}(e_1 = e_2) P$



式の表示

- 式の抽象構文 $e \in \text{Exp}$
 - $n \in N$
 - $e ::= n \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 * e_2 \mid e_1 / e_2$



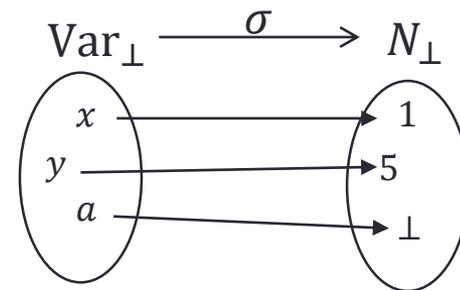
- $\mathcal{E}[[n]] = n$
- $\mathcal{E}[[e_1 + e_2]] = \mathcal{E}[[e_1]] + \mathcal{E}[[e_2]]$
- $\mathcal{E}[[e_1 - e_2]] = \mathcal{E}[[e_1]] - \mathcal{E}[[e_2]]$
- $\mathcal{E}[[e_1 * e_2]] = \mathcal{E}[[e_1]] \times \mathcal{E}[[e_2]]$
- $\mathcal{E}[[e_1 / e_2]] = \mathcal{E}[[e_1]] \div \mathcal{E}[[e_2]]$

変数付き式の表示

- 変数付き式の抽象構文 $e \in \text{Exp}$
 - $n \in N$
 - $v \in \text{Var}$
 - $e ::= n \mid v \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 * e_2 \mid e_1 / e_2$
- $e \in \text{Exp}$ の表示は現在の変数の値に依存する

状態 S

- $S = [\text{Var}_\perp \rightarrow N_\perp]$
- $\sigma \in S$ は変数を値に対応させる



Exp の表示

- $\mathcal{E} \in \text{Exp} \rightarrow [S \rightarrow N_\perp]$

- $\mathcal{E}[[n]]\sigma = n$
- $\mathcal{E}[[v]]\sigma = \sigma[[v]]$
- $\mathcal{E}[[e_1 + e_2]]\sigma = \mathcal{E}[[e_1]]\sigma + \mathcal{E}[[e_2]]\sigma$
- $\mathcal{E}[[e_1 - e_2]]\sigma = \mathcal{E}[[e_1]]\sigma - \mathcal{E}[[e_2]]\sigma$
- $\mathcal{E}[[e_1 * e_2]]\sigma = \mathcal{E}[[e_1]]\sigma \times \mathcal{E}[[e_2]]\sigma$
- $\mathcal{E}[[e_1 / e_2]]\sigma = \mathcal{E}[[e_1]]\sigma \div \mathcal{E}[[e_2]]\sigma$

文の表示

- 文の抽象構文 $c \in \text{Cmd}$

- $c ::= \text{null} \mid x := e \mid \text{if } (e_1 = e_2) c_1 \text{ else } c_2 \mid \text{while } (e_1 = e_2) c \mid c_1; c_2$

- Cmd の表示

- $\mathcal{C} \in \text{Cmd} \rightarrow [S \rightarrow S]$

$$S \xrightarrow{\mathcal{C}[[c]]} S$$

- $\mathcal{C}[[\text{null}]]\sigma = \sigma$

- $\mathcal{C}[[c_1; c_2]]\sigma = \mathcal{C}[[c_2]](\mathcal{C}[[c_1]]\sigma)$

- $\mathcal{C}[[c_1; c_2]] = \mathcal{C}[[c_2]] \circ \mathcal{C}[[c_1]]$

$$S \xrightarrow{\mathcal{C}[[c_1]]} S \xrightarrow{\mathcal{C}[[c_2]]} S$$

$$\underbrace{\hspace{15em}}_{\mathcal{C}[[c_1; c_2]]}$$

- $\mathcal{C}[[x := e]]\sigma = \sigma[\mathcal{E}[[e]]\sigma/x]$

$$\sigma[n/x][[y]] = \begin{cases} n & (y = x \text{ のとき}) \\ \sigma[[y]] & (\text{それ以外}) \end{cases}$$

- $\mathcal{C}[[\text{if } (e_1 = e_2) c_1 \text{ else } c_2]]\sigma = \text{cond}(\mathcal{E}[[e_1]]\sigma = \mathcal{E}[[e_2]]\sigma, \mathcal{C}[[c_1]]\sigma, \mathcal{C}[[c_2]]\sigma)$

文の表示(つづき)

- $\mathcal{C}[\text{while } (e_1 = e_2) c]\sigma$
- $\text{while } (e_1 = e_2) c \equiv \text{if } (e_1 = e_2) \{c; \text{while } (e_1 = e_2) c\} \text{ else null}$
- $\mathcal{C}[\text{while } (e_1 = e_2) c]\sigma = \mathcal{C}[\text{if } (e_1 = e_2) \{c; \text{while } (e_1 = e_2) c\} \text{ else null }]\sigma$
 $= \text{cond}(\mathcal{E}[e_1]\sigma = \mathcal{E}[e_2]\sigma, \mathcal{C}[c; \text{while } (e_1 = e_2) c]\sigma, \mathcal{C}[\text{null}]\sigma)$
 $= \text{cond}(\mathcal{E}[e_1]\sigma = \mathcal{E}[e_2]\sigma, \mathcal{C}[\text{while } (e_1 = e_2) c](\mathcal{C}[c]\sigma), \sigma)$
- $\mathcal{C}[\text{while } (e_1 = e_2) c] = \lambda\sigma. \text{cond}(\mathcal{E}[e_1]\sigma = \mathcal{E}[e_2]\sigma, \mathcal{C}[\text{while } (e_1 = e_2) c](\mathcal{C}[c]\sigma), \sigma)$
- $\mathcal{C}[\text{while } (e_1 = e_2) c]$ は次の式の不動点である
 - $\lambda w. \lambda\sigma. \text{cond}(\mathcal{E}[e_1]\sigma = \mathcal{E}[e_2]\sigma, w(\mathcal{C}[c]\sigma), \sigma)$
- $\mathcal{C}[\text{while } (e_1 = e_2) c] = \text{fix}(\lambda w. \lambda\sigma. \text{cond}(\mathcal{E}[e_1]\sigma = \mathcal{E}[e_2]\sigma, w(\mathcal{C}[c]\sigma), \sigma))$

$$\text{fix}(f) \equiv \sqcup_{i=0}^{\infty} f^i(\perp)$$

文の表示(まとめ)

- 文の抽象構文 $c \in \text{Cmd}$
 - $c ::= \text{null} \mid x := e \mid \text{if } (e_1 = e_2) c_1 \text{ else } c_2 \mid \text{while } (e_1 = e_2) c \mid c_1; c_2$
- Cmd の表示
 - $\mathcal{C} \in \text{Cmd} \rightarrow [S \rightarrow S]$

$$S \xrightarrow{\mathcal{C}[[c]]} S$$
- $\mathcal{C}[[\text{null}]]\sigma = \sigma$
- $\mathcal{C}[[c_1; c_2]] = \mathcal{C}[[c_2]] \circ \mathcal{C}[[c_1]]$
- $\mathcal{C}[[x := e]]\sigma = \sigma[\mathcal{E}[[e]]\sigma/x]$
- $\mathcal{C}[[\text{if } (e_1 = e_2) c_1 \text{ else } c_2]]\sigma = \text{cond}(\mathcal{E}[[e_1]]\sigma = \mathcal{E}[[e_2]]\sigma, \mathcal{C}[[c_1]]\sigma, \mathcal{C}[[c_2]]\sigma)$
- $\mathcal{C}[[\text{while } (e_1 = e_2) c]] = \text{fix}(\lambda w. \lambda \sigma. \text{cond}(\mathcal{E}[[e_1]]\sigma = \mathcal{E}[[e_2]]\sigma, w(\mathcal{C}[[c]]\sigma), \sigma))$

継続

- 副作用を扱うのが難しい

- $C[x := e]\sigma = \sigma[\mathcal{E}[e]\sigma/x]$

- 継続**とは

- 残りの計算を表す

- $C = [S \rightarrow S]$

- $\mathcal{C}[c]$

- 残りの計算を受け取り

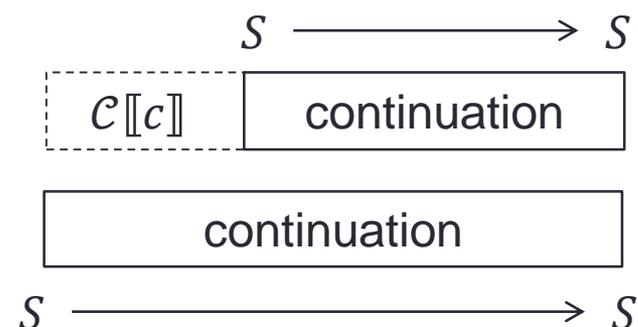
- c を含んだ計算を返す

- $\mathcal{C}[c] \in [C \rightarrow C]$

- $\mathcal{E}[e] \in [K \rightarrow C]$

- $K = [N_{\perp} \rightarrow C]$

- K は**式の継続**



$$[S \rightarrow S] \xrightarrow{\mathcal{C}[c]} [S \rightarrow S]$$

$$[N_{\perp} \rightarrow [S \rightarrow S]] \xrightarrow{\mathcal{E}[e]} [S \rightarrow S]$$

継続を使った文の表示

- 文の抽象構文 $c \in \text{Cmd}$

- $c ::= \text{null} \mid x := e \mid \text{if } (e_1 = e_2) c_1 \text{ else } c_2 \mid \text{while } (e_1 = e_2) c \mid c_1; c_2$

- Cmd の表示

- $C = [S \rightarrow S]$

$$[S \rightarrow S] \xrightarrow{\mathcal{C}[[c]]} [S \rightarrow S]$$

- $\mathcal{C} \in \text{Cmd} \rightarrow [C \rightarrow C]$

- $\mathcal{C}[[c]]\theta\sigma = \theta(\sigma')$

- 文 c を状態 σ で実行すると、状態が σ' となり、継続 θ に渡される

- $\mathcal{C}[[\text{null}]]\theta = \theta$

- $\mathcal{C}[[c_1; c_2]]\theta = \mathcal{C}[[c_1]](\mathcal{C}[[c_2]]\theta)$

- $\mathcal{C}[[x := e]]\theta = \mathcal{E}[[e]](\lambda n. \lambda \sigma. \theta(\sigma[n/x]))$

- $\mathcal{C}[[\text{if } (e_1 = e_2) c_1 \text{ else } c_2]]\theta = \mathcal{E}[[e_1]]\left(\lambda n_1. \mathcal{E}[[e_2]](\lambda n_2. \text{cond}(n_1 = n_2, \mathcal{C}[[c_1]]\theta, \mathcal{C}[[c_2]]\theta))\right)$

- $\mathcal{C}[[\text{while } (e_1 = e_2) c]] = \text{fix}\left(\lambda w. \lambda \theta. \mathcal{E}[[e_1]]\left(\lambda n_1. \mathcal{E}[[e_2]]\left(\lambda n_2. \text{cond}(n_1 = n_2, \mathcal{C}[[c]](w(\theta)), \theta)\right)\right)\right)$

継続を使った式の表示

- 式の抽象構文 $e \in \text{Exp}$
 - $n \in N$
 - $v \in \text{Var}$
 - $e ::= n \mid v \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 * e_2 \mid e_1 / e_2 \mid v ++$
- Exp の表示
 - $\mathcal{E} \in \text{Exp} \rightarrow [K \rightarrow C]$
 - $K = [N_{\perp} \rightarrow C]$
- $\mathcal{E}[[e]]\kappa\sigma = \kappa(n)(\sigma')$
 - 状態 σ で式 e 値を計算し, その結果 n を更新された状態 σ' とともに κ に渡す
- $\mathcal{E}[[n]]\kappa\sigma = \kappa(n)(\sigma)$
- $\mathcal{E}[[v]]\kappa\sigma = \kappa(\sigma[[v]])\sigma$
- $\mathcal{E}[[e_1 + e_2]]\kappa\sigma = \mathcal{E}[[e_1]]\left(\lambda n_1. \mathcal{E}[[e_2]]\left(\lambda n_2. \kappa(n_1 + n_2)\right)\right)\sigma$
- $\mathcal{E}[[v ++]]\kappa\sigma = \kappa(\sigma[[v]])\sigma[\sigma[[v]] + 1/v]$

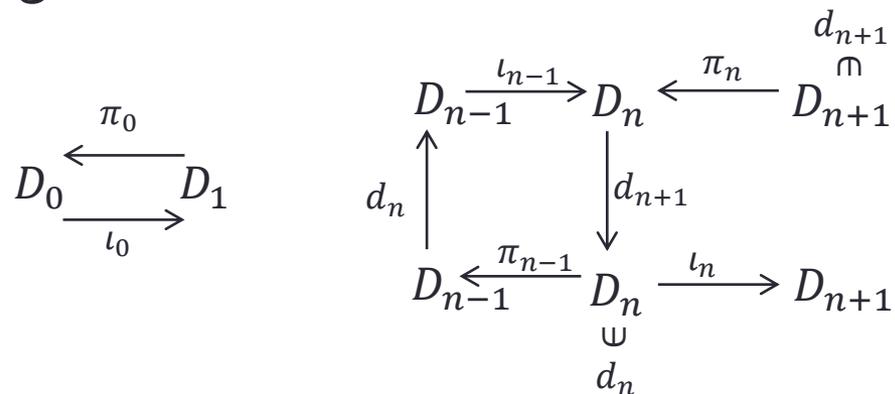
λ式の領域

- $D \cong [D \rightarrow D]$ となる領域 D を構成する

- $D_0 = \{\cdot\}_\perp$
- $D_1 = [D_0 \rightarrow D_0]$
- $D_2 = [D_1 \rightarrow D_1]$
- ...
- $D_{n+1} = [D_n \rightarrow D_n]$

- $D_0 \triangleleft D_1 \triangleleft D_2 \triangleleft D_3 \cdots \triangleleft D_n \triangleleft \cdots$

- $\pi_0 \in D_1 \rightarrow D_0 = [D_0 \rightarrow D_0] \rightarrow D_0$
- $\iota_0 \in D_0 \rightarrow D_1 = D_0 \rightarrow [D_0 \rightarrow D_0]$
- $\pi_n \in D_{n+1} \rightarrow D_n = [D_n \rightarrow D_n] \rightarrow D_n$
- $\iota_n \in D_n \rightarrow D_{n+1} = D_n \rightarrow [D_n \rightarrow D_n]$



$$\pi_0(d_1) = d_1(\perp_{D_0})$$

$$\iota_0(d_0) = \lambda x_1 \in D_0. d_0$$

$$\pi_n(d_{n+1}) = \pi_{n-1} \circ d_{n+1} \circ \iota_{n-1}$$

$$\iota_n(d_n) = \iota_{n-1} \circ d_n \circ \pi_{n-1}$$

- $D_\infty = \{ (d_0, d_1, \dots, d_n, \dots) \mid d_n \in D_n, \pi_n(d_{n+1}) = d_n \}$
 - $D_n \triangleleft D_\infty$
 - $D_\infty \cong [D_\infty \rightarrow D_\infty]$

λ式の表示

- λ式の抽象構文 $M \in \Lambda$
 - $x \in \text{Var}$
 - $M ::= x \mid \lambda x. M \mid M_1 M_2$
- $D_\infty \cong [D_\infty \rightarrow D_\infty]$
 - $\pi \in [D_\infty \rightarrow D_\infty] \rightarrow D_\infty$
 - $\iota \in D_\infty \rightarrow [D_\infty \rightarrow D_\infty]$
- λ式の表示
 - $\sigma \in S = \text{Var} \rightarrow D_\infty$
 - $\mathcal{L} \in \Lambda \rightarrow [S \rightarrow D_\infty]$
- $\mathcal{L}[[x]]\sigma = \sigma[[x]]$
- $\mathcal{L}[[\lambda x. M]]\sigma = \pi(\lambda v \in D_\infty. \mathcal{L}[[M]]\sigma[v/x])$
- $\mathcal{L}[[M_1 M_2]]\sigma = \iota(\mathcal{L}[[M_1]]\sigma)(\mathcal{L}[[M_2]]\sigma)$

$$\mathcal{L}[[Y]]\sigma = \pi(\lambda v \in D_\infty. \text{fix}(\iota(v)))$$

まとめ

- プログラミング言語の意味
 - 公理的意味
 - 操作的意味
 - 表示の意味
- 表示の意味
 - 意味関数(表示)
 - 継続
 - D_∞